APPLICATION FOR A UNITED STATES PATENT

UNITED STATES PATENT AND TRADEMARK OFFICE
(B&D Docket No. 301496.1005-118)

Date: December 30, 2003 Express Mail Label No. ER 094113975 US

Title: INTEGRATED PORTABLE ULTRASOUND IMAGING SYSTEM

Inventors: Alice M. Chiang, a citizen of United States of America and a resident of 4 Glenfeld East, Weston, Massachusetts 02493;
Jeffrey M. Gilbert, a citizen of United States of America and a resident of 1531 Liberty Street, Apt. 4, El Cerrito, California 04530; and
Steven R. Broadstone, a citizen of United States of America and a resident of 14 Hammond Place, Woburn, Massachusetts 01801.

Assignee: TeraTech Corporation
Burlington, Massachusetts

CROSS REFERENCES TO RELATED APPLICATIONS

This application is a continuation application of U.S. Application No. 09/921,976, filed on August 21, 2001, issuing on December 30, 2003, as U.S. Patent No. 6,671,227, which is a continuation application of U.S. Application No. 09/364,699, filed on July 30, 1999, which is a continuation application of International Application No. PCT/US98/02291, filed on February 3, 1998, now Publication No. WO 98/34294, which is a continuation-in-part application of U.S. Serial No. 08/965,663 filed on November 6, 1997, now U.S. Patent No. 6,111,816, issued August 29, 2000, which claims the benefit of U.S. Provisional Patent Application No. 60/036,387, filed on February 3, 1997, the entire teachings of the above applications being incorporated herein by reference.

BACKGROUND OF THE INVENTION

One use of sensor arrays is to isolate signal components that are traveling from, or propagating to, a particular direction. They find use in a number of different applications. For example, sonar systems make use of sensor arrays to process underwater acoustic signals to determine the location of a noise source; arrays are also used in radar systems to produce precisely shaped radar beams. Array processing techniques for isolating received signals are known as beamforming and when the same or analogous principles are applied to focus the transmission of signals, the techniques are referred to as beamsteering.

Considering the process of beamforming in particular, it is typically necessary to use a fairly large number of signal processing components to form the desired directional beams. The signal from each sensor is typically divided into representative components by subjecting each signal to multiple phase shift, or time delay, operations which cancel the equivalent time delay associated with the respective relative position of the sensor in the array. To form the directional beam the time shifted signals from each sensor are then added together. The imparted time delays are chosen such that the signals arriving from a desired angular direction add coherently, whereas those signals arriving from other directions do not add coherently, and so they tend to cancel. To control the resulting beamwidth and sidelobe suppression, it is typical for each time delayed signal to be multiplied or "amplitude shaded" by a weighting factor which depends upon the relative position of the sensor in the array.

Beamforming in one dimension can thus be realized through a relatively straight-forward implementation using a linear array of sensors and a beamforming processor, or beamformer, that delays each sensor output by the appropriate amount, weights each sensor output by multiplying by the desired weighting factor, and then sums the outputs of the multiplying operation. One way to implement such a beamformer is to use a tapped delay line connected to each array element so that the desired delay for any direction can be easily obtained by selecting the proper output tap. The beam steering operation then simply consists of specifying the appropriate tap connections and weights to be applied.

However, a beamforming processor becomes much more complex when a two dimensional sensor array is used. Not only does the number of time delay operations increase as the square of the size of the array, but also the physical structures required to connect each sensor to its corresponding delay becomes complex. At the same time, each delay unit must be provided with multiple taps for the formation of multiple beams. The problem can become prohibitively complicated when the simultaneous formation of multiple beams is required.

As to implementation choices, beamforming technology was originally developed for detection of acoustic signals in sonar applications. The beamformers built for these early sonars used analog delay lines and analog signal processing components to implement the sum and delay elements. Networks of resistors were then used to weight and sum the appropriately delayed signals. However, the number of beams that can be implemented easily with such techniques is limited since each beam requires many discrete delay lines, or delay lines with many taps and many different weighting networks. As a result, it became common to share a delay line by using scanning switches to sequentially look in all directions. However, with this approach only one beam is available at a given time.

Recent advancements in integrated circuit electronics has provided the capability to implement practical digital beamforming systems. In these systems a signal from each sensor is first subjected to analog to digital conversion prior to beamforming. The beamformers are implemented using digital shift registers to implement the delay and digital multiplier components to implement the required weighting. The shift registers and multiplier components are typically controlled by command signals that are generated in general purpose computers using algorithms or equations that compute the values of the delays and phase weightings necessary to achieve the desired array beam position. Beam control thus requires fairly complex data processors and/or signal processors to compute and supply proper commands; this is especially the case if more than one beam is to be formed simultaneously.

For these reasons, few multi-dimensional multiple beam systems exit that can operate in real time with a minimum implementation complexity.

SUMMARY OF THE INVENTION

The invention is a beamsteering or beamforming device (generically, a beamforming device), that carries out multi-dimensional beamforming operations as consecutive one-dimensional operations. In a preferred embodiment the two operations are interposed by a transpose operation. For example, beamforming for a two-dimensional array of sensors is carried out as a set of projections of each desired output beam onto each of the two respective axes of the array.

Signal examples are periodically taken from each sensor in the array and then operated on as a group, or matrix, of samples. A first one-dimensional (1D) beamformer is used to form multiple beams for each sensor output from a given row of the sample matrix. The multiple output beams from the first 1D beamformer are then applied to a transposing operation which reformats the sample matrix such that samples originating from a given column of the sensor array are applied as a group to second one-dimensional (1D) beamformer.

The beamformer can be implemented in an architecture which either operates on the samples of the sensor outputs in a series of row and column operations, or by operating on the sample matrix in parallel. In the serial implementation, a group of multiplexers are used at the input of the first 1D beamformer. Each multiplexer sequentially samples the outputs of the sensors located in given column of the array. The multiplexers operate in time synchronization such that at any given time, the outputs from the group of the multiplexers provide samples from the sensors located in each row of the array.

The multiplexers then feed the first 1D beamformer that calculates the projection of each row onto a first array axis, for each of the desired angles. In the serial implementation, the first 1D beamformer is implemented as a set of tapped delay lines formed from a series of charge coupled devices (CCDs). Each delay line receives a respective one of the multiplexer outputs. A number of fixed weight multipliers are connected to predetermined tap locations in each delay line, with the tap locations determined by the set of desired angles with respect to the first array axis, and the weights depending upon the desired beam width and sidelobe suppression. Each output of the first 1D beamformer is provided by adding one of the multiplier outputs from each of the delay lines.

The serial implementation of the transposer uses a set of tapped delay lines with one delay line for each output of the first 1D beamformer. The tapped delay lines have a progressively larger number of delay stages. To provide the required transpose operation, samples are fed into the delay lines in the same order in which they are received from the first 1D beamformer; however, the samples are read out of the delay lines in a different order. Specifically, at a given time, the output of the beamformer are all taken from a specific set of the last stages of one of the delay lines.

Finally, the second 1D beamformer consists of a set of tapped delay lines, fixed weight multipliers and adders in the same manner as the first 1D beamformer. However, the weights and delays applied by the second 1D beamformer are determined by the set of desired angles to be formed with respect to a second axis of the array.

In a parallel implementation of the invention, the multiplexers are not used, and instead the outputs of the array are fed directly to a set of parallel processing elements which operate on samples taken from all of the sensors simultaneously. Each processing element produces a set of beamformed outputs that correspond to the samples taken from one of the rows of sensors beamformed at each of the desired angles with respect to the first array axis. In this parallel implementation, the transposing operation is carried out by simply routing the outputs of the processing elements in the first 1D beamformer to the appropriate inputs of the second 1D beamformer. The second 1D beamformer likewise is implemented as a set of parallel processing elements, with each processing element operating on beamformed samples corresponding to those taken from one of the columns of the array, beamformed at each of the desired angles with respect to the second array axis.

In another preferred embodiment of the invention, a low power time domain delay and sum beamforming processor involves programmable delay circuits in sequence to provide a conformal acoustic lens. This electronically adjustable acoustic conformed lens has a plurality of subarrays that can be separately controlled to adjust viewing angle and their outputs coherently summed for imaging.

The invention provides a substantial advantage over prior art beamformers. For example, a device capable of steering up to one hundred beams for a ten by ten sonar array can be implemented on a single integrated circuit chip operating at a relatively low clock rate of 3.5 MegaHertz (MHZ), representing a continuous equivalent throughput rate of approximately 14 billion multiply-accumulate operations per second.

BRIEF DESCRIPTION OF THE DRAWINGS

This invention is pointed out with particularly in the appended claims. The above and further advantages of the invention may be better understood by referring to the following description in conjunction with the accompanying drawings, in which:

FIG. 1 is a block diagram of a serial implementation of a multiple simultaneous beamformer array according to the invention;

FIG. 2 illustrates the notation used herein to refer to various elements in the sensor array and their beam transformations;

FIG. 3 is a three-dimensional view of the array illustrating a beam formed therefrom together with projections of the beam onto a pair of orthogonal array axes;

FIG. 4 is a more detailed block diagram of the invention showing a first one-dimensional beamformer, a transposer, and a second one-dimensional beamformer together with various matricies of sensor samples formed by these components;

FIG. 5 is a block diagram of a serial implementation of a one-dimensional beamformer;

FIG. 6 is a block diagram of a serial implementation of the transposer;

FIG. 7 is a block diagram similar to that of FIG. 1, but showing a parallel implementation of the multiple simultaneous beamforming system;

FIG. 8 is a more detailed block diagram showing the interconnection of components for the parallel implementation;

FIG. 9 is a detailed block diagram of a processing element used in the parallel implementation;

FIG. 10 is a block diagram of a beamsteering system making use of the invention; and

FIG. 11 is a block diagram of a higher dimensional implementation of the multiple simultaneous beamforming system.

FIG. 12 illustrates a beamforming process utilizing programmable delays, shading and weighting.

FIG. 13 illustrates an array of subarrays for an electronically adjustable acoustic conformal lens in accordance with the invention.

FIG. 14A illustrates an electronically controlled beamforming system in accordance with the invention.

FIGS. 14B and 14C illustrate additional preferred embodiments of a beamforming system in accordance with the invention.

FIG. 15 illustrates a method of adjusting delays and shading in a scan pattern in accordance with the invention.

FIG. 16 illustrates a method for computing a method for computing delay requirements in parallel to afford real time processing in accordance with the invention.

Figs. 17A-17E are schematic diagrams of systems in accordance with the invention.

Figs. 18A-18B illustrate transmit beamforming systems in accordance with the invention.

Fig. 19 is a schematic illustration of an integrated circuit controller element in accordance with the invention.

Fig. 20 is another preferred embodiment of a channel controller.

Figs. 21A-21D illustrate sequential and parallel beamforming and associated image plane scan patterns.

Figs. 22A-22C illustrate image plane scan patterns for single processor, two processor and four processor systems.

Fig. 23 is an example illustrating sequential and parallel beamforming in a transducer array having 192 elements.

Fig. 24 illustrates a method of forming a Doppler sonogram in accordance with the invention.

Fig. 25 illustrates a method of forming a color flow map over time using a two dimensional array.

Fig. 26 illustrates a block diagram for a pulsed Doppler ultrasound imaging system.

Fig. 27 illustrates a two-PDP system for color flow mapping.

Fig. 28 is a process flow diagram for color flow map analysis based on a fourier transform process.

Fig. 29 is a process flow diagram for a color-flow map based on an optimal mean velocity.

Fig. 30 is a process flow diagram for a color-flow map based on a cross-correlation procedure.

The foregoing and other objects, features and advantages of the invention will be apparent from the following more particular description of preferred embodiments of the invention, as illustrated in the accompanying drawings in which like reference characters refer to the same parts throughout the different views. The drawings are not necessarily to scale, emphasis instead being placed upon illustrating the principles of the invention.

DETAILED DESCRIPTION OF THE INVENTION

Turning attention now to the drawings, FIG. 1 illustrates a system 10 for use with a two-dimensional, planar sensor array according to the serial implementation of the invention. The system 10 shown in FIG. 1 is a beamforming system, that is, system 10 operates with sensors 12 that detect received signals. However, as will be understood from the following description, the invention also applies to beamsteering systems where the sensors 12 are transmitting signals, and the signal directions are reversed.

The beamforming system 10 consists of a number of sensors 12 arranged in a planar array 14, a number, n, of multiplexers, 17-0, 17-1...17-(n-1), a first one-dimensional (1D) beamformer 18, a transposer 20, and a second 1D beamformer 22.

The array 14 consists of a number of sensors 12 arranged in an array of m rows 15-0, 15-1, 15- (m-1), each row having n sensors 12, and n columns 16-0, 16-1, 16-(n-1) having m sensors 12. The array may or may not be square, that is, n may or may not be equal to m.

The exact type of sensor 12 depends upon the particular use of the system 10. For example, in a system 10 intended for application to sonar, each sensor 12 is a hydrophone. In an application to radar systems, each sensor 12 is an antenna.

The remainder of the components of the system 10 operate to form multiple output beams 24 simultaneously. Before proceeding with a detailed description of the structure and operation of system 10, it is helpful to define a notation to refer to the various sensors 12 and as shown in FIG. 2. In particular, each of the (n x m) sensors 12 in the array 14 are referred to herein with respect to its relative horizontal and vertical position (x,y). Thus, for example, the notation Dx,y represents the signal provided by the sensor 12 located at column number x, row number y.

The notation Dx,v is used to refer to a beam formed using all of the sensors located in a given column, x, at a particular angle, v, with respect to array 14. Dw,y indicates a beam formed at a particular angle, w, using the sensors 12 in a given row y at an angle w with respect to the array. The notation Dw,v denotes the beam formed at a two dimensional angle (w,v) with respect to the array 14. Dw,v[t] indicates a beam formed at angles (w,v) at a time, t, or a depth, t, from the (x,y) plane of the array 14.

With reference now to FIG. 3, the operation of the invention may be better understood. In particular, FIG. 3 illustrate the planar array 14 in a three-dimensional view and an exemplary beam 26 formed on an angle (w,v) with respect to the (x,y) plane in which the array 14 is located. An additional third axis, z, is shown and is defined as being orthogonal to the axis (x,y) plane.

As can be seen from the illustration, the beam 26 formed at the angle (w,v) can be considered as having a pair of components projected upon two planes formed by the z axis and each of the array axes x and y. In particular, the beam 26 has a first component 26 –1 in the xz plane forming an angles w with respect to the x axis, as well as a second component 26-2 in the yz plane forming an angle v with respect to the y axis.

This assumption that the beam 26 can be represented as a pair of components 26-1 and 26-2 projected onto the orthogonal planes xz and yz is based upon an assumption that a far field operation approximation is valid for processing signals received from the array 14. The far field approximation will be valid for an array 14 in most sonar applications, for example. In such applications, the sensors 12 may typically be spaced approximately one meter apart, with the sound source being located at a distance of 100 meters or farther away from the array 14. Therefore, the far field approximation assumption is valid in applications where the sensor spacing, 1, is much smaller than the distance from the source being sensed. A difference of at least two orders of magnitude between the array sensor spacing and the distance to the source is sufficient for the approximation to be valid.

The operations required to form a number of desired beams 26 at a number of angles (w,v) can thus be decomposed into a pair of successive one-dimensional operation on the sensor outputs. Beam steering in a given direction (w,v) is accomplished as the projection of the beam 26 onto the xz plane forming an angle w with the x axis, followed by a projection onto the yz plane forming an angle v with respect to the y axis.

Returning now to FIG. 1, it can be seen that the first one-dimensional (1D) beamformer 18 thus performs the projection of the beam onto the xz plane, and that the projection onto the yz plane is performed by the subsequent operations of the second 1D beamformer 22. The first 1D beamformer 18 forms a set of beams for each desired beam angle v by taking samples of the signals output from the sensors 12 located in a given 15 row of the array 14. The outputs from the first one-dimensional beamformer 18 are reordered by the transposer 20, to arrange the beamformed outputs derived from a given column 16 together. This permits the second 1D beamformer 22 to perform the required operations at each of the angles w, while at the same time permitting the second 1D beamformer 22 to have the same internal structure as the first 1D beamformer 18.

FIG. 4 is a more detailed view of the operation of the first 1D beamformer 18, the transposer 20, and second 1D beamformer 22. It is assumed that the number of beams to be formed is equal to the number of sensors 12, such that a two dimensional array of beams 26 at each angle

(wi, vj) for I = o to n-1
and for j = 0 to m-1

is to be formed. However, it should be understood that the beams need not be the same dimension as the array, and indeed, the total number of beams 26 need not be the same as the total number of sensors 12.

The Figure 4 shows the inputs from the sensors 12 arranged as sample matrices. In particular, the samples input to the first 1D beamformer 18,

(Di,j) for I = o to n-1

and for j = 0 to m-1

are signal samples taken from the sensors located at each respective column and row position. The samples are operated on in two-dimensional groups, or matrices, by the system 10; the set of samples taken from the array 14 is referred to as the matrix 30.

The first 1D beamformer 18 performs a beam forming operation along the x direction at each of the desired beam angles w0, w1, ..., w(n-1). For example, the output Dw0, y0 represents the result of beamforming at a beam angle w0 the samples having a row coordinate of zero. That is, the output Dw0, y0, indicates the result of the beamforming operation on samples D0,0; D0,1; ..., D0, (n-1) located in row 15-0 at one of the desired beam angles w0. Likewise, Dw1, y0 corresponds to the output of the 1D beamformer 18 at beam angle w1,and so on.

The first beamformed matrix 32 output by the first 1D beamformer 18 thus represent input samples Dx,y beamformed along the x axis with each of the respective desired beam angles w0, w1, ..., w(n-1).

The transposer 20 transposes the rows and columns of the first beamformed matrix 32 to produce a transposed matrix 34. The transposed matrix 34 arranges the beamformed samples having the same corresponding y value located in a given column, and the beamformed samples having the same beam angle, w, located in a given row. This permits the second 1D beamformer to perform the 1D beamform operation on the samples in each row, with different angles vj, for j = 0 to (m-1).

As a result, the output matrix 36 from the second 1D beamformer 22 represents the two-dimensional beamformed outputs 24, with the output Dw0,v0 representing the beam at angle (w0,v0), the output Dw0,v1 corresponding to the beam angle (w0,v1), and so on. In other words, the sample outputs from the second 1D beamformer 22 correspond to all two dimensional beams formed the desired angles

(wi,vj) for I = 0 to n-1, and
for j = 0 to m-1.

Although FIG. 4 illustrates 1D beamformer 18 that translating along the x axis before 1D beamformer 22 translates along the y axis, it should be understood that the order of the 1D beamforming operations can be interchanged.

For the serial pipelined implementation of the invention, the matrices in FIG. 4 can also be interpreted as showing the time sequence in which the samples are operated upon by the first 1D beamformer 18, transposer 20, and second 1D beamformer 22. For example, in FIG. 4, each column in the matrix 30 represents samples taken from a particular one of the n multiplexers 17 in FIG. 1. Therefore, the multiplexers 17 in FIG. 1 sequentially select the output of one of the elements in a given column 13. For example, the first multiplexer 17-0 sequentially samples the outputs of the sensors 12 located in the first column 16-0 of the array 14. Likewise, a second multiplexer 17-1 sequentially samples the output of the sensors 12 located in a second column 16-1 of the array 14. The multiplexers operate sequentially such that each sensor 12 is effectively sampled at a sampling rate of fs.

The leftmost column of the matrix 30 indicates the order of the outputs taken from the first multiplier 17-0 of FIG. 1. The time sequence begins at the bottom row of the matrix 30 and works up the columns. Therefore, the samples are output by multiplexer 17-0 in a sequence Dx0,y0; Dx0,y1; ...; Dx0,ym-1. Since the multiplexers 17 operate in parallel and in row synchronization, the samples taken from the sensors 12 located on the first row, 15-0, that is, samples Dx0,y0; Dxy1,y0; Dx2,y0; ..., Dx(n-1),y0 are applied at the same time to the first 1D beamformer 18. Next, the samples from the second row 15-1, or samples Dx0,y1; Dx1,y1;..., Dx(n-1),y1 are applied to the first 1D beamformer 18, and so on.

Since the first 1D beamformer 18 performs a 1D beamforming operation on the samples in a given row 15, the first 1D beamformer can be implemented as a pipelined device such that a new row of samples can be immediately applied to the device and the operation repeated.

FIG. 5 is a more detailed view of the serial pipelined implementation of the first beamformer 18 for a 10 by 10 sensor array 14. The illustrated 1D beamformer 18 consists of ten tapped delay lines 40-0, 40-1, ..., 40-9, with a tapped delay line 40 being connected to the output from a corresponding one of the multiplexers 17-0, 17-1, ..., 17-9. A first group of multipliers 41-0-0, 41-0-1, ..., 41-0-9 are associated with the first tapped delay line 40-0, a second group of multipliers 41-1-0, 41-1-1, ..., 41-1-9 are associated with the second tapped delay line 40-1, and so on. The notation Dx0 above the first tapped delay line 40-0 indicated that the "x0" samples, that is, samples Dx0,y0; Dx0,y1; ...; Dx0,y9 are applied to the input of tapped delay line 40-0 in a repeating sequence.

The tapped delay lines 40 insert appropriate delays in the sensor outputs to account for relative propagation delays of a signal from a particular location. The delay lines 40 are each tapped such that the outputs from a certain number of delay stages are provided to the input of a multiplier 41.

The internal clock rate of each delay line 40 is ten times the input sample rate, fs, to permit the sampling of ten sensors into each tapped delay line 40. The total number of stages in each delay line 40 is sufficient to provide the maximum delay associated with forming a beam at the maximum required one of the angles, w. In the illustrated implementation, the total length of the delay line 40-0 shown is approximately 1350 stages, with ten tap positions set to provide 10 equally spaced apart angles, w. The position of the taps, that is the exact positions at which the inputs to the respective multipliers 41 is taken, depends upon the desired number of beams. The desired beam shape is defined by the weights applied to the multipliers 41.

Thus for an array 14 forming ten beams from each row 15 of input samples, the first 1D beamformer 18 consists of ten tapped delay lines, each delay line having ten taps and ten multipliers 41.

If the number and position of the desired beams is known in advance, the tap positions and constant values input as weights to the multipliers 41 can be hard wired or mask programmable.

The tapped delay lines 40 are preferably implemented as charge coupled device (CCD) type delay lines with fixed weight multipliers. A preferred implementation of this invention uses a non-destructive sensing type of charge domain device described in a co-pending United States patent application serial number 08/580,427, filed December 27, 1995 (MIT Case Number 7003), by Susanne A. Paul entitled "Charge Domain Generation and Replication Devices" the entire contents of which is hereby incorporated by reference.

The outputs of the multipliers 41 are then summed to accomplish the desired multiple simultaneous beam forming functions. The weighted outputs from the multipliers 40 are then simultaneously summed to form the desired beam output along a given row. For example, the output Dw0 is taken by summing the outputs of the last multipliers 41-0-9, 41-1-9, ..., 41-9-9 associated with each of the tapped delay lines 40.

FIG. 6 shows a detailed block diagram of pipelined serial implementation of the transposer 20. In the implementation for a 10 by 10 array, the transposer 20 contains ten parallel inputs which after ten consecutive output sample time periods produces a transposed 10 by 10 transposed matrix 34 according to FIG. 4. In this implementation for serial operation each of the 10 inputs Dx0, Dx1,...Dx9 are fed to a simple tapped delay line 50-0, 50-1, 50-2...50-9. The tapped delay lines 50 operate at the same clock rate, fs, as the input sampling rate.

The number of delay stages within each of the delay lines 50 progressively increases as the column index. For example, the first tapped delay line 50-0 has a length which is one more than the number of rows, m, in the matrix, or 11 stages, the second delay line 50-1 is 12 stages long and so on until the 10th delay line 50-9 is 20 stages long. Only the last 10 stages of each delay line 50 are tapped to provide for outputs.

In operation, the taps associated with each delay line are enabled at the same time in a time slot associated with that delay line. For example, at a first time p0 all of the taps from the first delay line 50-0 are enabled in parallel to provide the ten output Dw0,y0; Dw0,y1; ... Dw0,y9. At a second time p1, only the tap from the second delay line 50-1 are enabled. The operation continues until a time p9 at which the taps on the last delay line 50-9 are enabled.

FIG. 7 is a block diagram of a parallel implementation of the multiple simultaneous beamforming system 10. As can be seen from the illustration, the arrangement of the array 14 is similar to the serial implementation of FIG. 1. However, in this implementation the multiplexers are eliminated and all n x m sensor outputs are fed directly to the first 1D beamformer 118. The first 1D beamformer 118 parallel transposer 120 and second one-dimensional beamformer 122 in a manner analogous to the serial implementation, in that they perform the same matrix operations of FIG. 4. However, unlike the serial implementation, the parallel implementation replicates hardware.

FIG. 8 shows a more detailed view of the parallel implementation for 10 x 10 array. The one hundred samples Dx0,y0; Dx1,y0; ...; Dx9,y9 from the sensors 12 are fed directly to a bank of ten processing elements 140-0, 140-1, 140-2, 140-3,...140-9. A given processing element 140, such as processing element 140-0, receives samples Dx0,y0; Dx1,y0; ..., Dx9,y0 from a particular row 15-0 of the array 14; the processing element 140-0 thus provides the ten beamformed samples Dw0,y0; Dw1,y0; ...;Dw9,y0 for that row.

The ten processing elements 140 thus operate in parallel to produce 100 outputs at the same time, Dw0,y0; Dw1,y0; ...; Dw9,y9 that represent the ten respective beams formed outputs along the x axis.

In this parallel implementation, the transposer 20 is simply the proper routing of the outputs of the first 1D beamformer 18 to the inputs of the second 1D beamformer 22. The second 1D beamformer 122 is implemented in much the same manner as the first 1D beamformer 118 and includes a bank of ten processing elements 142-0, 142-1...142-9. The ten processing elements 142 operate in parallel to produce the 100 beamformed outputs Dw0,v0; Dw1,v1; ...; Dw9,v9.

An exemplary parallel processing element 140-0 is shown in detail in FIG. 9. Similar to the serial implementation, the parallel processing element 140-0 consists of ten tapped delay lines 144-0; 144-1, ..., 144-9 operating in parallel that insert delays at the sensor outputs to account for the relative propagation delays of the signals from a particular location. The sensor outputs are multiplied by a weighting factor and then summed. As in the previous embodiment, different weights are applied to different taps of a given delay line 144-0 by a set of multiples 146-0-0, 146-0-1, ..., 146-0-9. As for the serial implementation, in order to accomplish multiple beamforming simultaneously, multiple taps are incorporated along each delay line 144. The weighted tap outputs are then summed by a set of adders 148-0, 148-1, ..., 148-9 to form a particular beam output.

In this parallel implementation the clock rate of the delay lines 144 to accomplish real time processing may be ten times slower, for example, the clock rate need only be the same as the input sampling rate fs. However, the trade-off is that ten of the processing elements 140 are required to produce the necessary beamformed matrix 32.

Processing elements 142 associated with the second 1D beamformer 122 are similar to the exemplary processing element 140-0.

FIG. 10 shows a beamsteering implementation of the invention. Here, sensors 12 are transmitting devices, and the sample flow is in the reverse direction. Also, the multiplexers 17 are replaced by demultiplexers 117. Otherwise, the operation of the beamforming device 10 is analogous to the operation of the beamforming device of FIG. 1.

Finally with respect to FIG. 11 the invention can be adapted to provide higher dimensional beamforming or beamsteering. For example, a three-dimensional beamformer 200 may be implemented from a first 1D beamformer 218, a first transposer 220, and second 1D beamformer 222 as before, together with a second transposer 224 and third 1D beamformed 226. The third axis for beamforming may be an axis which is provided by a set of sensors arranged as a three-dimensional array.

Another preferred embodiment of the invention relates to a time-domain delay-and-sum beamforming processor that can simultaneously process the returns of a large two dimensional transducer array. The lower-power, highly integrated beamformed is capable of real-time processing of the entire array and enables a compact, affordable unit suitable for many different applications. A delay-and-sum beamformer allows a 2D array to "look" for signals propagating in a particular direction. By adjusting the delays associated with each element of the array, the array's directivity can be electronically steered toward the source of radiation. By systematically varying the beamformer's delays and its shading along a 2D image plane, a 2D scan response of the illustrated array can be measured and resulting 2D images representing the 2D radiation sources can be created.

A schematic diagram of a time-domain beamforming device for a 3D ultrasound/sonar imaging system 300 is illustrated in Figure 12. The system can provide continuous real-time large area 2D scanned images throughout a field of view at 30 frames per second or more. The image size is entirely programmable which can be either 128 by 128 or 256 by 256, for example. The delay-and-sum beamforming approach enables target range information to be obtained from the time-of-flight calculations. When a target area is identified by the electronically steerable imaging system, the beamforming electronics can be adjusted to zoom-in to a smaller field-of-view for high-resolution imagery. Furthermore, for a given peak transmit intensity, a matched filter 312 can be applied to the beamformed outputs to provide additional sensitivity.

As shown in Figure 12, preamplifier time-gain control 302 and broadband sampling 304 are performed on the transducer output signals. Programmable delaying 306, shading 308 and summing 310 are performed to generate the beamformed outputs. After match filtering 312, the resulting 2D image can be displayed 314.

The use of coded or spread spectrum signaling has gained favor in the communications community. It is now routinely used in satellite, cellular, and wire-line digital communications systems. In contrast, the application of this technique to acoustic systems has been prevented primarily due to signal propagation conditions and the relatively slow speed of sound in water (1500 m/s) or air when compared with electromagnetic propagation.

Despite these difficulties, the benefits of using coded signals in underwater acoustic systems, for example, offers the potential for higher-resolution imaging while significantly lowering the probability of external detection. These signals also provide signal processing gain that improves the overall system detection sensitivity.

Direct sequence modulation is the modulation of a carrier signal by a code sequence. In practice, this signal can be AM (pulse), FM, amplitude, phase or angle modulation. It can also be pseudorandom or PN sequence comprised of a sequence of binary values that repeat after a specified period of time.

The processing gain realized by using a direct sequence system is a function of the signal transmitted compared with the bit rate of the information. The computed gain is the improvement resulting from the RF to information bandwidth tradeoff. Using direct-sequence modulation, the process gain is equal to the ratio of the RF-spread spectrum signal bandwidth divided by the information rate in the baseband channel, $G_p=BW_{RF}/R$, where R is typically expressed in bits/Hz for digital communications.

The objective of a beamforming system is to focus signals received from an image point onto a transducer array. By inserting proper delays in a beamformer to wavefronts that are propagating in a particular direction, signals arriving from the direction of interest are added coherently, while those from other directions do not add coherently or cancel. For a multi-beam system, separate electronic circuitry is necessary for each beam.

Using conventional implementations, the resulting electronics rapidly become both bulky and costly as the number of beams increases. Traditionally, the cost, size, complexity and power requirements of a high-resolution beamformer have been avoided by “work-around” system approaches which form a number of transducer elements typically used in the sonar array. A typical configuration uses a center beam together with four adjacent beams aimed left, right, above and below the center. The beams are each formed from fifty or more elements in an array each phased appropriately for the coherent summation in the five directions of interest. The advantage of using so many elements is narrower beamwidths when compared with a smaller array, however, knowledge of the outside world is still based on a five pixel image. For real-time 3D high-resolution sonar imaging applications, a preferred embodiment utilizes an electronically steerable two-dimensional beamforming processor based on a delay-and-sum computing algorithm.

A delay-and-sum beamformer allows 2D array to "look" for signals propagating in particular directions. By adjusting the delays associated with each element of the array, the array's "look" direction or field or view can be electronically steered toward the source of radiation. By systematically varying the beamformer's delays and its shading or apodization along a 2D imaging plane, a 2D scan response of the array can be measured and resulting images representing the 2D radiation sources can be generated. To realize such a delay-and-sum beamformer, a programmable delay line is needed at each receiver. However, as the array is scanning through the imaging plane, there are two difficult implementation issues: first, each delay line has to be long enough to compensate for the path differences of a large area array, and second, the delay value has to be adjusted at each clock cycle for proper beam steer (i.e., the time-of-flight from the radiation source to the focal point has to be calculated at every clock cycle). For example, for a 10m range requirement with a resolution of one to two centimeters dictates an array aperture in the range of 40cm. To realize a thirty degree scanning volume, a maximum delay of 70 μs. This implies that a 2,300-stage delay line and a 12-bit control word are needed at each receiver to achieve the time-of-flight delay requirements. The long delay and large number of digital I/Os would set an upper limit on how many processors can be integrated on one chip. For example, for a 64-channel time domain beamforming electronics, a straightforward implementation would require 64 2,300-stage delay lines and 768 digital I/O pads. Such a large area chip and large number of I/O connections would make the implementation impractical.

An electronic beamforming structure is described to circumvent the impractically long delay line requirement and a delay-update computation based on the determination of time-of-flight surface coordinates is presented to reduce the digital I/O requirement. This electronic programmable beamforming structure functions as an electronically adjustable/controllable virtual acoustic lens. For this reason, this device is referred to herein as an electronically-controlled conformal lens.

An electronically-adjustable acoustic conformal lens uses a divided surface of a 2D transducer array in which plane "tiles" of relatively small subarrays are provided. As depicted in the embodiment of Figure 13, the tiles/subarrays 320 are made small enough so that when an object is placed within the field-of-view of the imaging system, the incident radiation 322 from the object toward each "tile" can be treated using a far-field approximation. Additional delay elements are incorporated within each subarray to allow all subarrays to be coherently summed (i.e., global near-field beamforming can be achieved by delaying and then summing the outputs from all subarrays). The delay-and-sum beamformer allows each subarray 324 to "look" for signals radiating from a particular direction as illustrated by the differences between configuration 324a and configuration 324b. By adjusting the delays associated with each element of the array, the array's viewing angle direction can be electronically steered toward the source of radiation. The delay line requirement for each element in the sub-array can be as short as several hundred stages. Only a single long delay for global summing is needed on each subarray processor.

A detailed diagram of an electornically-controlled beamforming system in accordance with the invention is shown in Figure 14A. This system consists of a bank of parallel sub-array processors 330_1 to 330_N. Each processor is a single integrated silicon circuit that comprises two components: a 2D sub-array beam former 332 for far-field beamsteering/focusing and an additional delay line 334 to allow hierarchical near-field beamforming of outputs from each subarray. As previously mentioned, the delays associated with each receiver element have to be adjusted to allow the subarray to "look" for signals arriving from a particular direction. As can be seen in Figure 14A, for an m-element sub-array, m-parallel programmable tapped delay lines 340_1 to 340_m are used for delay adjustment. Within each delay line, a time-of-flight computation circuit 342 is used to select the tap position output from a charge-domain circuit that non-destructively senses the tapped-delay line output. Inaccuracy of this charge sensing mechanism is only limited by the charge transfer inefficiency which is less than 10^{-6}. As a result, the delay can be dynamically adjusted at every clock cycle where the delay resolution is determined by the clock rate. Except for the clock skew, which can be controlled to less than 1ns, there are no other spurious or dispersive effects. Each receiver has a multiplier

344 for beam shading/apodization. Within each processor, all the multipliers share a common output 346. The summed charge is then applied to a second tapped delay line 350, which allows the delay from each subarray be adjusted so that all the subarrays can look at the same source of radiation. A charge-domain A/D converter 352 is used so that hierarchical summing 354 can be output digitally.

Shown in Figs. 14B and 14C are systems for 2D sonar beamformer with downconversion. The first embodiment shown in Fig. 14B depicts the downconversion following the matched filter 345. A complex-valued multiply 347 is performed, followed by low-pass filtering 353 and sample-rate down-conversion 355. The absolute magnitude is then taken to retrieve the envelope of the signal. The A/D conversion can follow the complex-valued multiplication, however, this embodiment uses an A/D in each Hierarchical Nearfield BF block.

The down converter of Fig. 14C is shown as the first operation 357 in each channel of a submodule. Although this can be a preferred method to reduce the signal-bandwidth/data rates through the remainder of the system, it is a more hardware intensive system. The multiplier 361 generates in phase (I) and quadrature (Q) components 363, 365 that are low pass filtered 367, converted 369 and summed 371 prior to delay 358.

By systemically varying the beamformer's delays and its shading along a 2D imaging plane, a rectilinear 2D scan pattern 360 of the array can be measured and resulting 2D images representing the 2D radiation sources can be created, see Figure 15. The system can provide continuous real-time large area scanned images throughout a large field of view at 30 frames/s or more. The delay-and-sum beamforming system provides target range information to be obtained from the time-of-flight calculations. When a target area is identified by the electronically steerable sonar system, the beamforming electronics 364 can be adjusted to zoom-in to a smaller filed-of-view for high-resolution imagery. Furthermore, for a given peak transmit intensity, a matched filter can be applied to the beamformed outputs to provide additional sensitivity. A low-power, finite-impulse-response (FIR) filter can be used to implement the matched filter at the output of the beamforming process to improve the system signal to noise ratio.

In real-time imaging applications, focus-and-steer images require knowledge of the time of flight from each source to each receiver in an array. To compute a new point on any time-of-flight surface requires finding the square root of the sum of squares, which is a computationally intensive task. A delay-update computation method can be used which reduces the determination of the rectangular coordinates of a new point on any time-of-flight surface to the computation time of a single addition. It is well-known that the method of moments can be used to synthesize basis functions that represent an arbitrary multidimensional function. Although the complete basis requires the determination of infinitely many coefficients, a finite-degree basis function can be generated using a least-mean-square (LMS) approximation. The specific form of the finite-degree basis depends on functional separability and limits of the region of support. Using the forward-difference representation of the truncated moments basis, a new functional value can be computed at every clock cycle. If the computation is performed within a square region of support, the direction of the finite difference corresponds to the direction that the function is computed. For example, functional synthesis from the upper-right to lower-left corners within the region of support implies the computation of a multidimensional, backward difference representation. Conversely the multi-dimensional, forward-difference representation, presented above, allows functional synthesis to proceed from the lower-left to the upper-left corners within the region of support. This approach produces images at least an order of magnitude faster than conventional time-of-flight computation.

In practice, the complete moments basis representation of a surface can be degree-limited for synthesis. One truncation method is to approximate f(x,y) with a bivariate polynomial of degree M. The bi-Mth degree approximation can be written as

$$\hat{f}(x,y)=\sum_{p=o}^{M}\sum_{q=o}^{M}\hat{a}_{p,q}x^{p}y^{q}$$

where $\hat{a}$ can be derived based on the LMS criterion,

$$\frac{\partial}{\partial a_{p,q}}\int_{x_1}^{x_2}\int_{y_1}^{y_2}[f(x,y)-\hat{f}(x,y)]^{2}dx\ dy=0.$$

Once the coefficients $\hat{a}_{p,q}$ of the bi-Mth degree polynomial $\hat{f}$ (x,y) possess positive-integer powers of x and y, it can be formulated as a stable, forward-difference equation. In general, $(M + 1)^2$ forward-difference terms are sufficient to describe a polynomial whose highest degree in x and y is M. The terms completely specify $\hat{f}$ (x,y) within its region of support.

Based on the assumption that the surface is to be scanned in a raster fashion and has been scaled, the step size is 1. For this case, the first and second forward differences in one dimension are

$$\Delta_x^1 = \hat{f}(x_0+1, y_0) - \hat{f}(x_0, y_0),$$

$$\Delta_x^2 = \hat{f}(x_0+2, y_0) - 2\hat{f}(x_0+1, y_0) + \hat{f}(x_0, y_0)$$

Using these forward differences, a second-degree polynomial in one dimensional can be written in difference form as

$$\hat{f}(x_0+k, y_0) = \hat{f}(x_0+y_0) + \left\lfloor \frac{k-1}{k} \right\rfloor \Delta_x^1 + \left\lfloor \frac{k-2}{k} \right\rfloor \Delta_x^2$$

where

$$\left\lfloor \frac{k}{n} \right\rfloor = \frac{k!}{n!(n-k)!}$$

It follows that the two-dimensional forward differences can be obtained by evaluating the cross product term in $\hat{f}$ (x,y),

$$\Delta_x^n \Delta_y^1 = \sum_{p=0}^{n} \sum_{q=0} (-1)^{n+1-p-q} \left\lfloor \frac{p}{n} \right\rfloor \left\lfloor \frac{q}{1} \right\rfloor \hat{f}(x_0+p, y_0+q)$$

A CMOS computing structure can be used to perform functional synthesis using the forward-difference representation of a multidimensional, finite-degree polynomial. This implementation allows the synthesis of arbitrary functions using repeated additions with no multiplications. An example of this computing structure 370 is presented in Figure 16 for a two dimensional, first-degree, forward difference realization. As shown in Figure 16 each register 372, represented by a rectangular box, contains the appropriate forward-difference term. Switches, which are located between registers, determine whether the x or y direction is synthesized. The advantage of this structure is that it allows additions to occur simultaneously at each of the adders 376. Thus only one addition time is required to produce the next function value. For a multi-channel processor, each channel contains its own functional update circuitry. As the beam is steered through a given imaging plane, the delay requirements for each channel are computed in parallel and can be updated within one clock period. For a 64 channel beamforming processors, at a 40 MHz clock rate, a continuous delay update rate of 30 billion bits/s can be achieved based on this approach.

Using this approach, instead of the alternative 11 bits/channel, the digital connectivity can be reduced to 1 bit-channel followed by on-chip computation circuitry to generate the equivalent 12 bit value while maintaining the 30 billion bits/s parallel update rate.

Preferred elements of a high performance ultrasound imaging system includes the ability to provide features such as 1) multi-zone transmit focus, 2) ability to provide different pulse shapes and frequencies, 3) support for a variety of scanning modes (e.g. linear, trapezoidal, curved-linear or sector), 4) multiply display modes such as M-mode, B-mode, Doppler sonogram and color-flow mapping (CFM). Preferred embodiment for such a system are based on the integrated beamforming chip described herein. All five systems can provide the desired capabilities described above, with different emphasis on physical size and power consumption.

In the system 400 shown in Fig. 17A, integrated circuits (modules) for beamforming 414, transmit/receive selection 416 and a preamplifier/TGC chip 418 are fully integrated within the probe-housing 402 with the transducer array 420, as is the system controller 422. The system controller 422 maintains proper clocking and operation of the memory 424 to assure continuous data output and also generates clock and control signals to indicate the intended recipient (among the three modules) of data packets at the memory output port. The controller 422 also interfaces with the host computer 406 (a generic personal computer, PC) via PCI bus or Firewire 426 along interface 404 to allow the host to update on-probe memory or to receive ultrasound data from the probe. (All signals pass between the host PC and probe via PCI or Firewire.) The tasks of signal down conversion, scan conversion (reformatting for display in a Cartesian coordinate system) and post signal processing are performed by microprocessing system 412 of the host PC. Additionally, in our system design, color-flow map and Doppler sonogram computations can be performed by two different implementations: a hardware-based implementation and a software implementation as shown in Fig. 17A. It is important to note that a dedicated Doppler-Processor chip can be mounted on a back-end (within the PC) card 408 and be used as a co-processor to the host computer to accomplish the Doppler sonogram computation and CFM computation. However, Fig. 17A depicts an implementation where the CFM and sonogram computations are performed by the host PC in software and output to display 410.

Fig. 17B depicts a system 440 that allows a more compact probe housing 442 or scanhead. In this design, the transducer array 444 is mounted in a probe housing 442 connected to a dedicated processing module 446 via coaxial cable 448. The component modules (beamforming, preamp/TGC and transmit/receive chips) are housed in the overall processing module 446, which communicates with the host PC 452 via PCI or Firewire 450. Multiple-beamforming is provided by this system 446. Control and synchronization is performed by the system controller located in the processing module.

Charge-domain processors 470 (CDP) for beamforming can also be fully integrated into a dedicated system, as shown in Fig. 17C. In this, the transducer array is housed in a separate scanhead unit 466 and connected to the host using coaxial cables 464. The suite of MCDP processing modules 470 (transmit/receive, preamp/TGC and beamforming chips are physically housed within the main system unit 462. This design supports multiple-beam beamforming with use of parallel CDP beamforming chips. This system covers the case in which beamforming tasks for an existing ultrasound system can be performed by CDP devices by replacing the original beamforming modules with their CDP equivalents.

A preferred embodiment for a compact scanhead that minimizes noise and cable loss is shown in Fig. 17D. This system 480 integrates the transmit/receive chip and preamp/TCG chip on the probe 482 with the transducer array. The system controller, memory and beamforming chip (or chips for multiple beamforming) are housed in a separate processing module 486 connected via PCI or Firewire to the host PC 488 which performs down conversion, scan conversion and post signal processing. This design reduces the size of the scanhead probe, as compared to the design in Fig. 17A.

The semi-integrated front-end probe 482 described in Fig. 17D, where the transmit/receive chip and preamp/TGC chip are placed on the probe with the transducer array, is coupled with a cable 484 to module 486 that uses CDP beamformers. This design compares to that in Fig. 17C, which describes use of CDP beamforming in an ultrasound system. The difference is that here additional processing is performed on the scanhead, reducing noise and cable losses relative to the system of Fig. 17C where all processing is performed after data are transmitted via coaxial cable from the scanhead to the host. System 490 of Fig. 17E retains the same or similar probe design as Fig. 17D, however the elements of the processing module 486 of Fig. 17D have been included in the processing system 492 of Fig. 17E.

The multi-dimensional beamformer processing system is a time-domain processor that simultaneously processes the transmit pulses and/or returns of a two-dimensional array 502. For transmit beamforming, the system can be used either in a bi-static mode, utilizing a separate transmit transducer array 502, or it can use the receive array 504 for transmit focus as well. As shown in Fig. 18A, for the bi-static configuration 500, the separate transmitter 502 can be a single-ping implementation that illuminates the whole image plane 506 with a single transmission. Alternatively, transmission can be implemented in a sparsely packed beam pattern that covers the image plane 514 as shown in Figure 18B. For transmit beamforming, a transmit control chip is needed for providing delays to the high-voltage driving pulses applied to each transducer element of array 512 such that the transmitted pulses are coherently summed on the image plane at the required transmit focus point 516.

The multi-channel transmit/receive chip performs the functions of transmit beamforming, switching between transmit receive modes (TRswitch), and high-voltage level shifting. As shown in Fig. 19, the multi-channel transmit/receive chip consists of, a global counter 542 which broadcasts a master clock and bit values to each channel processor. A global memory 544 which controls transmit frequency, pulse number, pulse sequence and transmit/receive select. A local comparator 546 which provides delay selection for each channel. For example, with a 60-MHz clock and a 10-bit global counter, the comparator can provide each channel with up to 17-*ms* delay. A local frequency counter 548 which provides programmable transmit frequency. A 4-bit counter provides up to sixteen different frequency selections. For example, using a 60-MHz master clock, a 4-bit counter can be programmed to provide 60/2=30MHz, 60/3-20MHz, 60/4=15MHz, 60/5=12 MHz, 60/6=10MHz and so on. A local pulse counter 550 which provides different pulse sequences. For example, a 6-bit counter can provide programmable transmitted pulse lengths from one pulse up to 64 pulses. A locally programmable phase selector 552 which provides sub-clock delay resolution. For example, for a 60-MHz master clock and a two-to-one phase selector provides 8-ns delay resolution.

While typically the period of the transmit-chip clock determines the delay resolution, a technique called programmable subclock delay resolution allows the delay resolution to be more precise than the clock period. With programmable subclock delay resolution, the output of the frequency counter is gated with a phase of the clock that is programmable on a per-channel basis. In the simplest form, a two-phase clock is used and the out put of the frequency counter is either gated with the asserted or deasserted clock. Alternatively, multiple skewed clocks can be used. One per channel can be selected and used to gate the coarse timing signal from the frequency counter. In another implementation 560 shown in Fig. 20, the T/R switch and the high-voltage level shifter 562 are separated from the other components to allow high-voltage operation.

By systematically varying beamformer delays and shading along a 2D imaging plane, a 2D scan response of a 2D transducer array can be measured and resulting 2D images representing the 2D radiation sources can be created. This method can be extended to scan not just a 2D plane but a 3D volume by systematically changing the image plane depth as time progresses, producing a sequence of 2D images, each generated by the 2D beamforming processors as described above. The sequence of images depicts a series of cross-section views of a 3D volume as shown in Fig. 21A. In this manner a complete scan of a 3D object can be obtained. There are two modes of operation: sequential or parallel. In sequential mode, a single steerable beamforming processor is used with the 2D array. As shown in Fig. 21B, the image plane is serially scanned pixel by pixel (i.e. the beamforming is computed pixel-by-pixel) until the whole image plane is processed. In parallel mode, more than one beamforming processor is used. Fig. 21C depicts the case of two steerable beamforming processor. At any given time, two receive beams are formed, one by each processors. The corresponding scan pattern is shown in Fig. 21D where the image plane is divided into two halves and each beamforming processor is used to scan half of the image plane. Consequently, the frame rate can be doubled in this parallel beamforming mode. A scan pattern generated by four parallel receive beamforming processors is depicted in Figs. 22A-22C. It can be deduced that the frame rate for the four parallel-beam beamforming system can be four times faster than that of a single-beam beamforming system. In general, for an *m*-parallel receive beam system, the frame rate can be increased by a factor *m*.

The same sequential vs parallel receive beamforming architecture is applicable to a 1D linear or curved linear array. Fig. 23 shows, as an example, a 192-element one dimensional array 600. In sequential mode, with a single 64-element beamforming processor used, the scan lines (the received beams) are formed one by one. That is to say, line 1 at 602 is formed first by processing returned echoes from elements 1 through 64 at a view angle –0 (angles defined with respect to the normal direction from the transducer face). Line 2 is formed next by processing returns from elements 64 through 128 at a normal view angle. Line 3 is then formed by returns from elements 1 through 64 at a view angle –0 + 0/*S*, and so forth. It follows then that the (S-1)th line is formed by returns from element 1 through 64 at a view angle –0/*S*. The Sth line is formed by returns from elements 128 through 192 at the normal direction from the transducer face. Finally the Lth scan line is formed at 604 by processing the returns from elements 128 through 192 at a view angle +0. In parallel mode, multiple beamforming processors are used in parallel. For the case of two 64-element beamforming processors, at any given time, two scan lines (or two beams) are formed by the first processor and the odd numbered lines by the second processor. The number of scan lines is selected based on the imaging quality requirement of a given application. For example, an imaging system can be designed to provide 256 scan lines, e.g. L=256 and W=64. On the other hand, for a high-resolution imaging application, a system can be designed to provide 1024 scan line, e.g., L=1024, and S=128. It is important to note that the frame rate of the two-processor system can be twice as fast as that of a single-processor system. In general, for an *m*-processor beam forming system, the frame rate can be increased by a factor of *m* relative to a single-processor system.

A Doppler sonogram 620 can be generated using single-range-gate Doppler processing, as shown in Fig. 24. The operation of this method is as follows. A sequence of *N* ultrasonic pulses is transmitted at a pulse repetition frequency f_{prf} along a given view angle. The return echoes 622 are range gated and only returns from a single range bin are used, meaning that only the returned signals corresponding to a region a selected distance (e.g. from depth *d* to *d+Sd*) from the transducer array along the selected viewing angle are processed to extract Doppler information. The velocity profiles of scatterers in the selected region can be obtained by computing the Doppler shifts of the echoes received from the scatterers. That is, Fourier transformation at 624 of the received time-domain signal provides frequency information, including the desired Doppler shifts, f_d. The velocity distribution of the scatterers in the region of interest can be obtained from the relationship

$$fd = [\frac{2v}{c}]f_c,$$

where *c* is the speed of sound in the transmitting medium and f_c is the center frequency of the transducer. As an example, if $N = 6$ and $f_{prf} = 1$ KH$_z$, the above equation can be used to generate a sonogram displaying 16 ms of Doppler data. If the procedure is repeated every N/f$_{prf}$ seconds, a continuous Doppler sonogram plot can be produced.

The following relates to a pulse-Doppler processor for 3D color flow map applications. The pulsed systems described here can be used for interrogating flow patterns and velocities, such as the flow of blood within a vessel. The time evolution of the velocity distribution is presented as a sonogram, and different parts of the vessel can be probed by moving the range gate and varying its size. The ultimate goal for probing the circulatory system with ultrasound is to display a full map of the blood flow in real time. This enables the display of velocity profiles in vessels and the pulsatility of the flow. One step toward meeting this goal is to use color flow mapping (CFM) systems. They are an extension of the multigrated system described in the above paragraph, as the blood velocity is estimated for a number of directions (scan lines) in order to form an image of flow pattern. The velocity image is superimposed on a B-mode image, and the velocity is coded as color intensity and direction of flow is coded as color. For example, a red color indicates flow toward and blue flow away from the transducer. A color-flow map based on pulsed Doppler processing is shown here in Fig. 25. Instead of a single range bin, data from *J* range bins, which correspond to returns from different depths, are processed in parallel. At a given scan angle, after *N*-pulse returns at array 640 are processed, the outputs represent *JX*N range-vs-Doppler distribution. For a given range bin, e.g., a given depth, the data can be used to generate a *N* velocity distribution profile. Mean velocity calculated from this distribution profile can be used to produce one point on the color-flow map of a given depth, the standard deviation can be used to assess turbulence. If the procedure is repeated every scan angle such that the whole image plane has been covered, a 3D color-flow map which depicts as a sequence of 2D color-flow plots, each one corresponding to different depths at d_1, d_2 and d_3 can be produced.

Algorithms can be used to compute the first moment and the velocity distribution of the pulse returns. Instead of a Fourier transform-based computation, a cross correlation technique, described in Jensen, Jorgen A., "Estimation of Blood Velocities Using Ultrasounds", Cambridge Univ. Press, 1996, the entire contents of which is incorporated herein by reference, can also be used to produce a similar color flow map. Furthermore, an optimal mean velocity estimation can be used.

Mean velocity (i.e., first spectral moment) estimation is central to many pulse Doppler data processing. With applications such as Color Flow Map for displaying mean velocity, inherent requirements for high-scan rate and fine (azimuth) scan patterns restrict the allocation of pulse samples to but a small number per range cell. As a result, these applications operate at times near the fundamental limits of their estimation capabilities. For such specific needs, an optimal Doppler Centroid estimation in the case of known spectral width (SW) and signal-to-noise ratio (SNR) is described.

Let us consider the usual probabilistic model for pulse-Doppler observation of a complex-valued vector return, $z_1, z_2, \ldots, z_N$ corresponding to a single range cell with N equally-spaced samples of a complex Gaussian process with covariance matrix T = E[ZZ*]. We also adopt the common single-source sample-covariance model consisting of Gaussian-shaped signal plus uncorrelated additive noise:

$$r_n = Se^{-8(\pi\sigma_v n\pi)^2} e^{-j4\pi\bar{v}n\tau/\lambda} + V_{noise}\delta_n \,(0 \le n\angle N)$$

where the model parameters $\bar{v}$ and σ_v represents mean Doppler velocity and Doppler SW, λ is the transducer RF wavelength, and S and N respectively represent signal to noise power magnitudes. Let us define

$$G = G(\sigma) = \begin{bmatrix} p(0) & p(1) & \cdot & p(N-1) \\ p(1) & p(2) & \cdot & p(N-2) \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ p(N-1) & p(N-2) & \cdot & p(0) \end{bmatrix}$$

$$where\, p(n) = e^{(\pi n\sigma)^2/2}, \sigma = \frac{\sigma_v}{v_\lambda}, and\, V_\lambda = \frac{\lambda}{4\pi}$$

In the case of maximum likelihood (ML) estimation, it results in a simple mean velocity expression

$$\omega_{ML} = \arg\min \mathrm{Re}(\sum_{n=0}^{N-1} r_n'^{\,r} e^{-jan})$$

$$where\, r_n^r\, is\, the\, weighted\, autocorrelation\, estimate\, defined\, by$$

$$r_m^r \equiv \sum_{i=0}^{N-n-1} Z_i^{*} \gamma_{i,i} + nZ_{i+n},$$

$$where\, \omega = \frac{\bar{v}\pi}{V_\lambda} and \gamma_{i,k}\, and\, the\, element\, of\, the\, matrix\, \Gamma,$$

$$where\, \Gamma = \Gamma(\sigma,\eta) = [SG + V_{noise}\ \Gamma]^{-1}$$

$$and\ I = diag[1 1 \cdots 1].$$

The generic waveform for pulse-Doppler ultrasound imaging is shown in Fig. 26. The waveform consists of a burst of N pulses 660 with as many range depth samples as needed are collected for each pulse in the burst. Fig. 26 also shows a block diagram of a signal processor for this imaging technique, where the returned echoes received by each transducer are sampled and coherently summed prior to in-phase and quadrature demodulation. The demodulated returns are converted to a digital representation, and then stored in a buffer memory until all the pulse returns comprising a coherent interval are received. The N pulse returns collected for each depth are then read from memory, a weighting sequence, υ(n), is applied to control Doppler sidelobes, and an N-point FFT 661 is computed. During the time the depth samples from one coherent interval are being processed through the Doppler filter, returns from the next coherent interval are arriving and are stored in a second input buffer. The FFT output can be passed on directly to a display unit or further processed by time-averaging Doppler prior to display.

The CDP device described here performs all of the functions indicated in the dotted box 662 of Fig. 26, except for A/D conversion, which is not necessary because the CDP provides the analog sampled data function. This CDP Pulsed-Doppler Processor (PDP) device has the capability to compute a matrix-matrix product, and therefore has a much broader range of capabilities. The device computes the product of two real-valued matrices by summing the outer products formed by pairing columns of the first matrix with corresponding rows of the second matrix.

In order to describe the application of the PDP to the Doppler filtering problem, we first cast the Doppler filtering equation into a sum of real-valued matrix operations. The Doppler filtering is accomplished by computing a Discrete Fourier Transform (DFT) of the weighted pulse returns for each depth of interest. If we denote the depth-Doppler samples *g(kj)*, where

$$g(k,j)=\sum_{n=0}^{N-1}v(n)f(n,j)\exp(-j2\pi kn/N)$$

The weighting function can be combined with the DFT kernel to obtain a matrix of Doppler filter transform coefficients with elements given by

$$w(k,n)=w_{k,n}=v(n)\exp(-j2\pi kn/N)$$

The real and imaginary components of the Doppler filtered signal can now be written as

$$g_{r,kj}=\sum_{n=0}^{N-1}(w_{r,kn}f_{r,nj}-w_{i,kn}f_{i,nj})$$

$$g_{r,kj}=\sum_{n=0}^{N-1}(w_{r,kn}f_{i,nj}+w_{i,kn}f_{i,nj})$$

In Eq.(4), the double-indexed variables may all be viewed as matrix indices. Therefore, in matrix representation, the Doppler filtering can be expressed as matrix product operation. It can be seen that the PDP device can be used to perform each of hte four matrix multiplications thereby implementing the Doppler filtering operation.

A block diagram of the PDP device is shown in Fig. 26. The device includes a J-stage CCD tapped delay line 64, J CCD MDACs 666 (multiplying digital-to-analog converters), *JXK* accumulators, *JXK* Doppler sample buffers 668, and a parallel-in-serial out (PISO) output shift register 670. The MDACs share a common 8-bit digital input on which elements from the coefficient matrix are supplied. The tapped delay line performs the function of a sample-and-hold, converting the continuous-time analog input signal to a sampled analog signal.

A two-PDP implementation for color flow mapping in an ultrasound imaging system is shown in Fig. 27. In this device, during one pulse repetition interval, the top PDP component computes all the terms of the form $w_r f_r$ and $w_s f_r$ as shown in Eq. 5, while the bottom component computes terms of the form -$w_i f_i$ and $w_r f_i$. The outputs of each component are then summed to alternately obtain g_r and g_i. Doppler and color flow map processing involves significant computation. This processing can be accomplished in software using a general-purpose microprocessor. The presence of instructions optimized for matrix-matrix operations, such as Intel MMX feature set, can substantially improve performance. A software flow chart for color-flow map computation 700 based on the Fourier transform computation is shown in Fig. 28. After initialization 702, the downconverted data is obtained 704 and the pointer P is at the beginning of the scan angle 706, the range data collected and stored 708, a weighting function is applied 710, the Fourier transform is computed 712, the magnitude *z(k)* is computed for each frequency followed by the computation of first and second moments 716 and displayed 718 in color. The pointer is incremented and each scan line is processed as needed.

A software flow chart 740 for color-flow map computation based on the optimal mean velocity estimation described above is shown in Fig. 29. After initialization 742 the downconverted data is obtained 744 and the pointer P is at the beginning of the scan angle 746, the range data is collected and stored 748, a weighting autocorrelation function 750 is computed based on the Equation (3). It follows then a mean velocity can be estimated 752 based on Equation (2). The mean velocity is displayed 754 in color. The pointer is incremented and each scan line is processed as needed.

A software flow chart for color-flow map computation based on the cross-correlation computation 760 is shown in Fig. 30. After initialization 762 the range data is obtained 766. The cross correlation is computer 768 and averaged 770. The velocity distribution 772, first and second moments 774 are computed and displayed 776. The range date is increased until all data on a given scan line are all processed. The process repeats for the next scan line until all the scan lines along a completed image plane are processed.

While we have shown and described several embodiments in accordance with the present invention, it is to be understood that the invention is not limited thereto, but is susceptible to numerous changes and modifications as known to a person skilled in the art, and we therefore do not wish to be limited to the details shown and described herein but intend to cover all such changes and modifications as are obvious to one of ordinary skill in the art.

While this invention has been particularly shown and described with references to preferred embodiments thereof, it will be understood by those skilled in the art that various changes in form and details may be made therein without departing from the spirit and scope of the invention as defined by the appended claims.